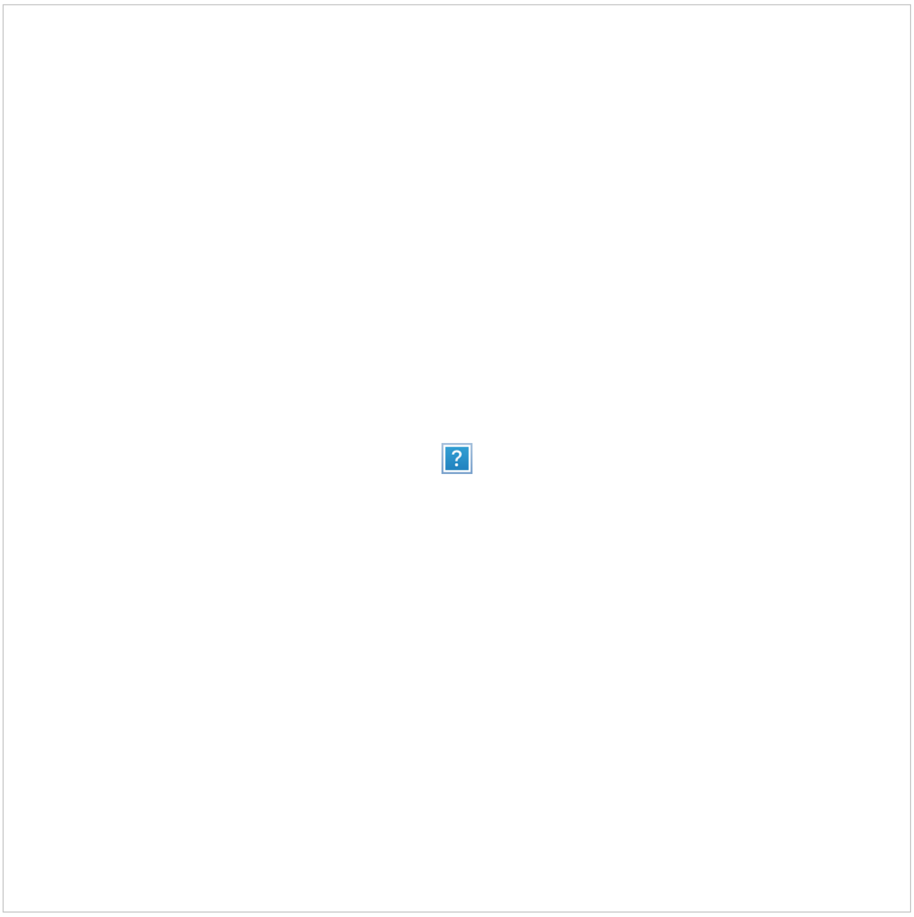
How the W3C Web of Things brings JSON Schema to the Internet of Things

Juan Cruz Viotti

2022-08-17 · 5 min read



Challenge

The Internet of Things (IoT) is the network of physical "smart" objects that exchange data with other devices over the Internet. While communication and interoperability are by definition the crux of the Internet of Things, the emergence of custom or proprietary solutions results in devices that cannot talk to each other due to differences in data interchange mechanisms.

To integrate these disparate devices, developers must work with a growing set of protocols, serialization formats and API specifications. This results in repetitive, non-scalable and error-prone work that is difficult to automate.

While technologies like OpenAPI and AsyncAPI largely solve this problem in the context of Web APIs, they fall short for describing networks of non-HTTP multi-protocol devices and do not consider different modes of interaction based on their meaning in the physical world.

Solution

To solve these problems, the W3C Web of Things works on providing standardized building blocks that make use of JSON Schema.

JSON Schema is used for validating abstractions of physical entities called Thing Descriptions, and to model and describe data sent by Internet of Things consumers and producers in a multi-protocol manner.

We are on-board with JSON Schema since Draft 4, and even the first draft versions were quite enough to model data models of Internet of Things devices. Currently, we are on Draft 7 and expect to move to 2020-12 or a newer one when starting our new charter in February 2023.

We observe JSON Schema becoming more and more stable over the years and Postman's support to the community gives us even more confidence on it. JSON Schema is delivering exciting features, and more importantly these features are becoming more and more consistent across implementations. For example, vocabularies are very promising and we plan to explore them further in conjunction with Semantic Web technologies.

Impact

The W3C Web of Things relies on JSON-LD for linking vocabularies and ontologies that already exist on the web, such as SAREF, Units of Measure and Schema.org. Being based on the JSON data model, JSON Schema fits well with JSON-LD when integrating both technologies as part of a single specification.

Adopting JSON Schema means that we do not need to invest effort on inventing our own schema language. As a consequence of the popularity of JSON Schema, we can often rely on existing tooling instead of implementing our own parsers, validators, code generators and UI generators for all the programming languages that our community might be interested in.

JSON Schema has proven to be a foundational block for creating higher-level tooling that is specific to our needs. As a notable example, the W3C Web of Things community has developed a tool that makes use of JSON Schema definitions inside Thing Descriptions to generate matching payloads for the purpose of stress and penetration testing.

Given the growing popularity of using JSON Schema to generate web-based forms, W3C Web of Things specifications can be used to automatically generate dashboards and user interfaces to interact with the objects modeled by Thing Descriptions.

Key Impact Results

The ultimate metric of success for a standard like the W3C Web of Things is adoption, and community adoption is not something you can just "engineer". Because we adopt JSON Schema, developers who are familiar with it have a smaller learning curve when learning the W3C Web of Things standard.

W3C Web of Things

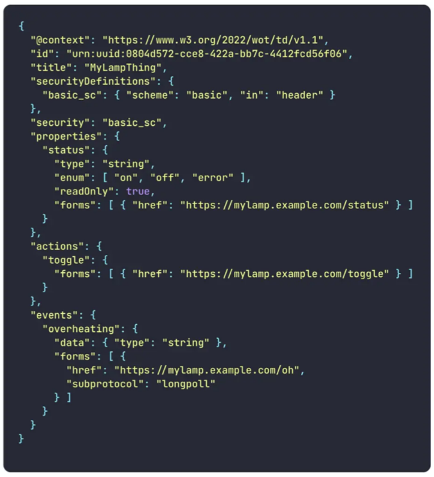
World Wide Web Consortium (W3C) is the standardization organization behind most of the standards used in the Web. We are the Web of Things Working and Interest Groups, who work on the standardization of Web of Things at the W3C.

We aim to create an interoperability layer so that different

Internet of Things platforms, protocols and standards can operate together.

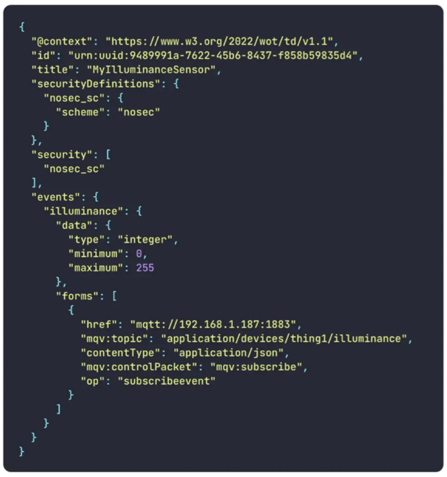
The work has started as a discussion in the Web of Things Community Group in 2013-2014. It has proceed to an Interest Group in 2015 that has collected the use cases and defined the standards to be worked on. Since 2016, the working group is working on different standards on the Web of Things with the first publications of the Thing Description and Architecture recommendations in 2019.

At the time of this writing, our Working Group consists of 96 participants representing 36 organizations and the our Interest Group consists of 131 participants representing 48 organizations.



```
{
  "@context": "https://www.w3.org/2022/wot/td/v1.1",
  "id": "urn:uuid:0804d572-cce8-422a-bb7c-4412fcd56f06",
  "title": "MyLampThing",
  "securityDefinitions": {
    "basic_sc": { "scheme": "basic", "in": "header" }
  },
  "security": "basic_sc",
  "properties": {
    "status": {
      "type": "string",
      "enum": [ "on", "off", "error" ],
      "readOnly": true,
      "forms": [ { "href": "https://mylamp.example.com/status" } ]
    }
  },
  "actions": {
    "toggle": {
      "forms": [ { "href": "https://mylamp.example.com/toggle" } ]
    }
  },
  "events": {
    "overheating": {
      "data": { "type": "string" },
      "forms": [ {
        "href": "https://mylamp.example.com/oh",
        "subprotocol": "longpoll"
      } ]
    }
  }
}
```

Example 1. A Thing Description that models an HTTP lamp API with a status property, a toggle action and an overheating event.



```
{
  "@context": "https://www.w3.org/2022/wot/td/v1.1",
  "id": "urn:uuid:9489991a-7622-45b6-8437-f858b59835d4",
  "title": "MyIlluminanceSensor",
  "securityDefinitions": {
    "nosec_sc": {
      "scheme": "nosec"
    }
  },
  "security": [
    "nosec_sc"
  ],
  "events": {
    "illuminance": {
      "data": {
        "type": "integer",
        "minimum": 0,
        "maximum": 255
      },
      "forms": [
        {
          "href": "mqtt://192.168.1.187:1883",
          "mqv:topic": "application/devices/thing1/illuminance",
          "contentType": "application/json",
          "mqv:controlPacket": "mqv:subscribe",
          "op": "subscribeevent"
        }
      ]
    }
  }
}
```

Example 2. A Thing Description that models an illuminance API with a sensor event that is transmitted over the MQTT protocol.

Getting Started

We invite you to leverage your existing JSON Schema knowledge to play with the Internet of Things through the use of our specifications and tools. You might already have a smart device next to you that does not come with a Thing Description!

If you write your own Thing Description, you can programatically interact with your device by using our Node.js Web of Things implementation, or play with it through our Node-RED integration.

To learn more, head over to our documentation, watch our introductory videos, have a look at the many examples provided as part of the Thing Description specification and try them out on the Thing Description Playground app.